

### **REMARKS**

By this response, claims 21-41 are pending. Claim 37 is amended while all others appear as previously presented. Substantively, the Examiner rejects claims 21-35 and 37-41 as obvious in view of U.S. Patent No. 6,470,332 to Weschler and U.S. Patent No. 5,491,817 to Gopal et al. Claim 36 is rejected as obvious over Weschler and Gopal in further view of U.S. Patent No. 6,260,039 to Schneck. The pending claims have many features and aspects not found in the combination of references, however, and reconsideration is respectfully requested.

For instance, each of the claims require the notion that an administrator utility is configurable to associate directory classes of two or more disparate directories so that users can search the disparate directories with a single query.<sup>1</sup> The utility also preferably resides with a directory shell and users search from a directory browser. Nowhere, however, do Weschler and Gopal alone or in combination teach this. Among other things, Weschler insists on multiple queries, instead of a single query, and Gopal insists on computing environments with search queries that “uniquely and unambiguously identif[y] the element in the network.” *Col. 2, ll. 62-63*. Weschler also insists on keeping results of a first of the many queries unreported to users so that further queries can be executed, e.g., “before any results are reported back for the first query, the subsequent queries proceed further down a ‘tree’ structure in relationship to further profiles relative to any candidate profiles matched initially to determine if they too match the subsequent query strings in the sequence of queries that has been specified [by the user].” *Col. 9, ll. 8-14*. Gopal, on the other hand, also insists on creating additional or linking directories, with pointers in between, so that indirect searching occurs in more than just the original to-be-searched directories involved

---

<sup>1</sup> Of course, each of the independent claims is worded slightly differently than the other and the scope of any given claim is governed by its express language.

with user queries. *E.g., col. 5, ll. 50-59.* Furthermore, Gopal severely criticizes techniques utilizing “administrative overhead” for performing acts of “relating directories.” *Col. 4, l. 23.* For at least these reasons, the claims are patentable over the art of record. For claim 36, the combination with Schneck is invalid for the substantial reasons already justified by the Applicant in its Pre-Appeal Brief Request for Review and sustained by the Notice of Panel Decision. Not the least of which relates to Schneck exclusively teaching searching within a single directory, typified by a root and variously sequenced nodes, as representatively seen in Schneck’s Figures 5-7.

With more specificity, Weschler concerns itself with a “system, method and computer program product for searching for, and retrieving, profile (or directory) attributes based on other attributes of the target profile and that of associated profiles.” *Abstract.* In concrete embodiments, this amounts, first, to “specification of a sequence of [plural] query strings.” *Col. 9, ll. 49-50.* It amounts, second, to executing a first of the queries and placing the results in a “set of candidates profiles.” *Col. 9, ll. 53.* From there, “the results are ‘candidates’ for the next query of the series.” *Col. 9, ll. 7-8.* The process then repeats one or more times, *e.g., steps 410 - 416, Figure 4.* Unlike the instant invention, this is clearly the instance of multiple queries and, iteration of same, to get search results. Certainly, this is not the limitation of a “single query” of the instant invention as the Examiner contends it is.

Also, Weschler gives a representative example of his process whereby a query labeled “Query 3” includes first “(TYPE=USER)” and second “(TYPE=NET)(EMAIL~=SUN).” *Figure 3.* During searching, however, it is the situation that after execution of the first query for TYPE=USER, “profiles 304, 306, 308 and 310 are all [potential] candidates” of the whole query. *Col. 9, ll. 32-33.* However, there are no search results for the user at this point in time and so the second or additional query is executed in order to eventually zero-in on the sub-profile labeled 316. Importantly, this means that even if a single query were used in

Weschler, inoperability is the result. That is, no search results are obtainable from a single query and such cannot then be useful in rendering the claims obvious.

Moreover, Weschler recites convenient examples with search terms that are clearly consistent with one another in terms of directory classes. Namely, Weschler unequivocally teaches a zip code as “ZIP” and a location as “LOCALE.” *E.g., Figure 3*. However, if one directory labeled the zip code as “ZIP” and another labeled it as “MAIL CODE,” or one directory labeled the location as “LOCALE” while another labeled it as “COUNTRY,” the convenience given by Weschler would be negated. That is why the instant invention<sup>2</sup> further provides “a directory shell” whereby an administrator can “configure” the “ZIP” of the one directory to the “MAIL CODE” of the other, or the “LOCALE” of the one directory to the “COUNTRY” of the other, into a “user-searchable category.” In turn, users can search the zip code of both the “ZIP” and the zip code of the “MAIL CODE,” or the location of both the “LOCALE” and the location of the “COUNTRY,” with “a single query” in “a directory browser” and find the appropriate zip code or location search results. Nowhere does Weschler teach this. Because Weschler also does not present any scenario other than a consistent “ZIP” or “LOCALE,” for example, Weschler does not appreciate the context of the instant invention. It cannot then even hint at the capabilities of the present invention. To this end, the Applicant agrees with the Examiner’s assertion that Weschler “does not clearly teach a directory browser with the directory shell whereby users can search the directory classes with a single query of the user-searchable category.” *E.g., 5-2-06 Office Action, p. 3, 3<sup>rd</sup> ¶*.

Gopal, on the other hand, does not supply the missing teaching rendering the claims obvious. Among other things, Gopal teaches a linking directory 300 in addition to the to-be-

---

<sup>2</sup> The claim language in this discussion comes from representative claim 21. Of course, the other claims are precisely defined by their own words and nothing is to be inferred.

searched directories, e.g., 75,<sup>3</sup> whereby the functionality of pointers, e.g., linking identifiers, serve to assist in various user searches. For the overall process to work, however, all the local directories 75 must relate somehow to the linking directory 300, including the linking application 200, and do so by way of the DSA 70, e.g., Figure 4.

In contrast, the instant invention<sup>4</sup> altogether avoids the Gopal functionality in favor of an administrator role whereby disparate directories are linked together in an administrator utility of a directory shell. In this manner, users can come along later without knowledge or care of the role of the administrator and conduct single queries of disparate directories. Administrator roles in Gopal, however, are discouraged and intentionally avoided. For instance, Gopal (1) recognizes the role of the administrator in the “seeAlso” mechanism of the X.500 standard and (2) characterizes it as a “drawback” requiring “high administrative overhead” for acts of “relating directory entries.” *Col. 4, ll. 22-26*. In turn, Gopal leaves behind the role of the administrator in favor of a “linking directory 300” and “linking application 200” that is used in addition to the to-be-searched directories 75. As a matter of law, Gopal cannot then be said to render the claims obvious, especially considering the role of the administrator utility in the instant invention whereby disparate directories are configured into a user-searchable category.

Similar to Weschler, Gopal also teaches its process in the context of conveniently and consistently defined search terms, not disparate directories like the instant invention needing association by way of an administrator utility so that users can search the multiple disparate

---

<sup>3</sup> The Applicant appreciates that element 75 relates to the admitted prior art of Gopal, and not necessarily the Gopal process. Since no single reference element is later referenced for the entirety of the Gopal searchable directories, only names, it is a matter of convenience to cite to element 75.

<sup>4</sup> The claim language in this discussion comes from representative claim 21. Of course, the other claims are precisely defined by their own words and nothing is to be inferred

directories with a single query from a directory browser. That is, Gopal references the search term “JOHN DOE” in both the background and the detailed description portions of his patent. Yet, “JOHN DOE” is mentioned as a “distinguished name” that is a “term of art” for conducting searching. *Col. 2, l. 60*. In turn, the user must know the name “JOHN DOE” in order to get its search results, e.g., the credit card balance, the account number, the business phone, etc. To the extent the Gopal searchable directories are not linked with the identifier, and to the extent a user simply desired to find “JOHN DOE,” it may be tedious because one directory might have “JOHN DOE” in a first class labeled “NAME,” while another might have “JOHN DOE” in a class labeled “IDENTITY.” With the instant invention, however, the user need not know the disparate directories having “NAME” and “IDENTITY” and tedious searches are avoided because the administrative role, via the administrator utility, associates the two together in a user-searchable category, such as “PERSON,” and users search the category from a browser with a single query. Gopal nowhere teaches this functionality. Because Gopal does not present any scenario other than the JOHN DOE scenario with linking identifiers, for example, Gopal does not then fully appreciate the context of the instant invention. It cannot be said to render obvious.

In combination, the Weschler and Gopal also certainly fail. In one instance, Weschler and Gopal do not individually appreciate the context of the present invention. In turn, their combination certainly cannot appreciate it and must fail as a rejection. In another instance, the Examiner points to the combination of references for the desired motivation of “enabling a user to retrieve information in response to queries from directory users, and allowing *a user* to access directory information about an object in one context *by knowing directory information about the object in another context.*” *Emphasis added, 5-2-06 Office Action, p. 4, 1<sup>st</sup> ¶.*

The Applicant agrees the law allows for examining the nature of the problem to be solved when determining motivation.<sup>5</sup> However, the instant invention does not address solving a problem where users, who conduct searching in disparate directories, “know directory information about the object in [multiple] context[s],” as contended. Rather, users of the instant invention have no idea about the disparity of the directories because an administrative role is used to associate the disparities together into a user-searchable category that users then search with a single query. In fact, users do not need to know that “JOHN DOE” in one directory is labeled a “NAME” while in another is labeled an “IDENTITY” (to continue the discussion example from above). All users need to know is searching occurs in a single user-searchable category. The Examiner, thus, identifies an erroneous motivation for combining the references and together they fail.

In still another instance, the combination of references fails for solving different problems. On the one hand, the instant invention came about from the unfulfilled need of a user to search multiple disparate directories with a single query of a user-defined category. For example, when two companies A & B join in a merger, each pre-merger company includes a human resources department with a directory listing of all the employees. When searching for information about an employee in one directory, a user would search the “Employee List” directory from company A and the “People Who Work Here” directory from company B. With the instant invention, the user can now search the merged company directory with a single query of a single category, such as “EMPLOYEES,” because of the administrative role that associates the two disparate directories together in an administrator utility.

---

<sup>5</sup> “A suggestion or motivation to modify prior art teachings may appear in the context of the public prior art, in the nature of the problem addressed by the invention, or even in the knowledge of one of ordinary skill in the art.” *Underlining added, Princeton Biochemicals, Inc. Beckman coulter, Inc.*, 411 F.3d 1332 (Fed. Cir. 1995).

From the Applicant's specification, this is given as disparate or "different" directories that "potentially have different names for class attributes." *Page 14, l. 27.* Representative Cisco and Novell corporate directories are given as disparate directories, including differing directory classes 51, that are searchable with a single query under an administrator-created user-searchable category 62 having the name "Find All." *Page 8, l. 13 - page 9, l. 4.* In turn, mapping or associating the differing directory classes 51 of the Cisco and Novell directories into a single user-searchable category 62 occurs, for example in a directory shell 60. In one embodiment, the directory shell 60 "includes two aspects: an administrator utility and a directory browser." *Page 11, ll. 14-15.* Then, the administrator utility allows the administrator to disable or enable searching on a directory class by various mechanisms, such as checking a box (or not) under an Enabled Column of a table 122, for example. *E.g., page 12, l. 30 et seq.* User searching occurs, for example, via a query portion 210 of the directory browser shown as a page 200 in Figure 8. Results of the search are displayed in a variety of panels 220, 230 on the same page. Formats for both the utility and browser are preferably HTML.

On the other hand, neither Gopal nor Weschler solve or attempt to solve this or similar problems. They both also fail to solve the same or similar problems as one another. That is, Gopal allows a user to access requested information by submitting known information about an object in one context along with a request for information about that object in another context. For example, a user has a need for a person's address but only knows the person's name, John Doe. The user searches the directory of names for John Doe to access a linking identifier attached to the record of John Doe's name. That linking identifier is then present on all other information relating to John Doe, including Doe's address, and a search for the linking identifier within the directory of addresses will provide the information requested by the user. In other words, Gopal accesses a single directory then uses information therein to

search additional directories. It also occurs via an additionally-created linking directory. The instant invention, however, searches multiple disparate directories, directly, without resorting to an intermediate or indirect directory as in Gopal. The invention in Gopal also relies on information already known by the user in order to obtain the linking identifier and further information relating thereto. The instant invention has no such reliance.

The Weschler reference does not solve the same problem as the present claims either. As before, Weschler is intended as a method of searching data, where the results of a first of a sequence of multiple queries are used as the candidates for the following query of the multiple of queries. The present invention, however, does not care about multiple queries, but a single query for searching disparate directories.

Even if the Gopal and Weschler references are properly combined, and the Applicant does not admit this, the combination does not result in the present claims. For instance, Weschler describes multiple search queries iteratively processed to get results. Gopal, on the other hand, teaches a linking directory, with linking identifiers, that is created in addition to the to-be-searched directories. In combination, the result yields multiple search queries iteratively processed by way of a linking directory having linking identifiers and such differs significantly from the pending claims.

Regarding claim 36, the Examiner combines Schneck with the earlier-described Weschler and Gopal combination. As submitted in previous papers, Schneck concerns itself with a "Web to X.500 Gateway" 100 that interfaces between either an administrative or user access point (in the form of an admin interface 106 or web browser 108, respectively) and a X.500 distributed system agent 104, storing items such as information about people, products or resources. In one instance, users search a directory of information to find people via their name, such as "Jane Doe," or title, such as "Manager." *Figures 15B, 15C*. In other instances, users search the directory to find "John Doe" via a search of an organization name,



such as “XYZCorp,” and an organizational unit, such as “Sales.” *Figures 5-7*. In this regard, various mapping schemes 212 are used.

At no time, however, does Schneck teach searching of disparate directories each having a directory class with dissimilar directory objects and data as required in claim 36 of the instant invention. Rather, Schneck exclusively teaches searching within a single directory, typified by a root and variously sequenced nodes, as representatively seen in its *Figures 5-7*. In turn, Schneck is unable (while the present invention is able) to search for people, for example, listed as “SALES” in Schneck’s one directory and “SALESPERSONS” in a second, completely disparate director. The present invention even enables it in a single query after administrators associate the “SALES” and the “SALESPERSONS” together in an administrator utility having a directory shell. As to Schneck’s searching in a single directory, the Applicant submits this is well known. It is also discussed generally in relation to directory hierarchy in the Applicant’s specification in the Background section at pages 2-4.

Because Schneck has already been sustained by the Panel in the Pre-Appeal Brief Request for Review as insufficient in rejecting the claims (in combination with Prompt), it appears the Examiner’s position of obviousness, for claim 36 especially, is nothing more than the cautioned-against selective culling of tidbits from the references in an attempt to fit the limitations of the claims. Not that the Examiner has fit the limitations, rejections of this sort are clearly discouraged under the law.<sup>6</sup>

---

<sup>6</sup>As is well established, “virtually all [inventions] are combinations of old elements.” *Ruiz v. A.B. Chance Co.*, 69 USPQ2d 1686, 1690 (Fed. Cir. 2004). Also, an obvious determination under 35 U.S.C. 103(a) requires an “as a whole” analysis of the prior art to otherwise prevent an impermissible “evaluation of the invention part by part.” *Id.* For otherwise, “an obviousness assessment might break an invention into its component parts (A+B+C), then find a prior art reference containing A, another containing B, and another containing C, and on that basis alone declare the invention obvious.” *Id.* In turn, “this form of hindsight reasoning, using the invention as a roadmap to find its prior components, would discount the value of combining various existing

Individually, each of the pending claims define over both Weschler and Gopal, as well as the other art of record, for one or more of the reasons given hereafter.

**Claim 21:** This claim requires a directory shell able to reference two or more disparate directories. The shell includes both an administrator utility and directory browser. The utility is configurable to “associate” the directory classes of the two or more disparate directories into “a user-searchable category.” The browser is then a structure where users search the directory classes of the two or more disparate directories with “a single query” of the user-searchable category. Weschler and Gopal, alone or in combination, do not render this obvious. Weschler insists on “multiple” queries and Gopal insists on avoiding administrative roles, especially in linking activities. For at least these reasons, the claims define over the art.

**Claims 22-27:** These claims depend from claim 21 and further require nuances of the system. Namely, they specify the whereabouts of the two or more disparate directories and the directory shell (claim 22); they describe structure that sends the single query and allows communication between the structure and the disparate directories (claims 23 and 24); they describe the category as including a category attribute mapped to class attributes of the directory class (claim 25); and they define where search results are displayed in the browser and where associating occurs in the utility (claims 26 and 27). Weschler and Gopal, however, fail as references in this regard, especially considering the limitations of parent claim 21.

**Claim 28:** This claim describes methodology of “providing” the directory shell to “enable” the administrator utility to associate directory classes of two or more disparate directories in a single user-searchable category. Searching of the category then occurs from

---

features or principles in a new way to achieve a new result - often the very definition of invention.”  
*Id.*

a directory browser of the shell. Searching is also done in a “direct” manner with a “single” query. Weschler, however, requires “multiple” queries. Gopal requires indirect searching by way of an additionally created “linking” directory which is mentioned as “a directory of directories.” *Col. 5, ll. 58-59.*

**Claims 29 and 30:** These claims relate to specifying how the utility of claim 28 is enabled to associate directory classes and to displaying search results in a panel of the browser. Neither Weschler nor Gopal intimate any such features, especially since these claims further include the features of claim 28.

**Claim 31** relates to computer readable media having executable instructions for performing the methodology of claim 28.

**Claim 32:** This claim is similar to claim 21 and is patentable for the reasons given above. However, it further builds on claim 21 and requires a table, a query portion and a panel in the utility and browser for associating, searching and displaying search results. Weschler and Gopal are far short of rendering this obvious. Neither reference teaches a table, a query portion, or a panel in the utility and browser for associating, searching and displaying search results.

**Claims 33-36:** More narrowly, these claims build on claim 32 and particularly specify panels for the browser, check boxes for associating, an enable column in the table and an HTML format for both the utility and browser.

**Claim 37:** This claim requires the creation of a single user-searchable category from directory classes of two or more disparate directories and an ability to “directly” search these classes with a “single query” of the category. Negatively, the claim further requires the absence of any creation or use of a virtual “or other” directory. Weschler clearly teaches multiple queries to satisfy various search results and, without a doubt, Gopal is quite

Application Serial No. 09/943,786  
Amendment dated July 7, 2006  
Reply to Office Action dated May 2, 2006

incapable of searching without further establishing or creating a "linking" directory 300. The combination of references must fail.

**Claims 38-40:** These claims all relate to how a category of claim 37 is created. In one aspect, creating further includes associating directory classes in the utility (claim 38). In another, it relates to creating more than one category (claim 39). In still another, it requires providing a directory shell for loading on a computer in communication with servers having the disparate directories (claim 40).

**Claim 41** relates to computer readable media having executable instructions for performing the methodology of claim 37.

Consequently, the Applicant submits all claims are in a condition for allowance and requests a timely Notice of Allowance be issued for same. *To the extent any fees are due, although none are believed due, the undersigned authorizes their deduction from Deposit Account No. 11-0978. Finally, the Applicant requests a change in the attorney document number of record. Namely, please replace NO078/100002 with 1363-007.* The docket number changed when the new Power of Attorney (POA) went into effect. The Patent Office indicated acceptance of the changed POA in a paper mailed on March 14, 2005.

Respectfully submitted,

**KING & SCHICKLI, PLLC**



Michael T. Sanderson  
Registration No. 43,082

247 North Broadway  
Lexington, Kentucky 40507  
(859) 252-0889

CERTIFICATE OF MAILING

I hereby certify that this correspondence  
is being sent by First Class Mail in an envelope addressed to  
Mail Stop Amendment, Commissioner for Patents,  
P.O. Box 1450, Alexandria, VA 22313-1450, on  
Date July 7, 2006

